This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended)   A system for automatically generating source code from a functional model comprising:

        a modeler for defining at least one of a plurality of code elements and a structure of a code block and generating a graphical representation <u>displayed on a computer</u> of the at least one code element and structure of the code block<u>, wherein the modeler processes input comprising a code block of source code from an innermost element to an outermost element and generates from the input a code model comprising a graphical representation of a structure and flow of the code block</u>.

2. (Original)   The system of claim 1, further comprising a user interface for receiving the definition of the at least one code element and the structure of the code block

3. (Original)   The system of claim 1, further comprising a selector for selecting at least one of a plurality of programming languages in which to generate the source code from the functional model.

4. (Original)   The system of claim 3, further comprising a code generator for receiving the graphical representation of the at least one code element and the structure of the code block and the at least one programming language and generating source code in each of the at least one programming languages.

5. (Original)   The system of claim 2, wherein the at least one programming language comprises C++.

6. (Currently Amended)   The system of claim 2, wherein the at least one programming languages comprises C#.

7. (Currently Amended)   The system of claim 2, wherein the at least one

programming languages comprises ~~Visual Basic~~ VB Script.

8. (Currently Amended)   A method of automatically generating source code from

a functional software model comprising:

processing a block of programming code from an innermost element to an outermost

element and generating from the processed block of programming code a functional software

model;

defining a plurality of code elements within ~~a~~ the block of programming code;

specifying a structure of the block of programming code including the plurality of

code elements; and

generating from the plurality of code elements and the structure of the block of

programming code including the plurality of code elements a graphical representation of the

code elements and flow of the block of programming code.

9. (Currently Amended)   The method of claim 8, further comprising ~~a user~~

~~interface for~~ receiving the definition of the plurality of code elements with the block of

programming code and ~~for~~ specifying the structure of the block of programming code via a

user interface.

10. (Original)   The method of claim 8, further comprising specifying at least one

target language in which source code for the graphical representation is to be generated.

11. (Original)   The method of claim 10, further comprising generating the source code

in the at least one target language.

12. (Currently Amended)   The method of claim 8, wherein ~~a~~ one of the plurality of

code elements comprises a variable, comment, constant, object, function, method, prototype,

member, data type, callback, delegate, reference, field, variant, property, interface, class, type, enumeration, structure, primitive, array, or event handle.

13. (Currently Amended)   The method of claim 8, wherein a one of the plurality of code elements comprises a code relation.

14. (Original)   The method of claim 13, wherein the code relation comprises a mathematical operator.

15. (Currently Amended)   The method of claim 8, wherein a one of the plurality of code elements comprises an evaluation entity.

16. (Original)   The method of claim 15, wherein the evaluation entity comprises one of a method call, a plurality of code entities, a plurality of code relations or an instantiation of a class.

17. (Currently Amended)   The method of claim 8, wherein a one of the plurality of code elements comprises a passive entity.

18. (Original)   The method of claim 15, wherein the passive entity comprises a comment or a modeling diagram.

19. (Currently Amended)   The method of claim 8, wherein a one of the plurality of code elements comprises a block entity.

20. (Currently Amended)   The method of claim 19, wherein the block entity comprises a method entity, a member entity, a class entity, a namespace entity or a file entity.

21. (Original)   The method of claim 20, wherein a many-to-many relationship exists between block entities.

22. (Currently Amended)   A computer-readable <u>storage</u> medium including computer-readable instructions for

processing a block of programming code from an innermost program element to an outermost program element and generating from the processed block of programming code a functional software model;

defining a plurality of code elements within <s>a</s><u>the</u> block of programming code;

specifying a structure of the block of programming code including the plurality of code elements;<u> and</u>

generating from the plurality of code elements and the structure of the block of programming code including the plurality of code elements a graphical representation of the code elements and flow of the block of programming code<u> comprising the functional software model</u>.